Implementation of automated tools for linear and non-linear homogenization : examples through the extension of the `MFront` code generator

A. MARTIN[a], T. HELFER[b], R. MASSON[c], O. FANDEUR[d]

a. CEA, DES, IRESNE, DEC, SESC, antoine.martin@cea.fr
b. CEA, DES, IRESNE, DEC, SESC, thomas.helfer@cea.fr
c. CEA, DES, IRESNE, DEC, SESC, renaud.masson@cea.fr
d. CEA, DES, ISAS, DM2S, SEMT, olivier.fandeur@cea.fr

Résumé :

Cet article présente le développement d'outils numériques pour l'implémentation de lois de comportement homogénéisées basées sur des schémas de type "champ moyen", en mécanique non-linéaire. Ces schémas font intervenir un système non-linéaire à l'échelle de la microstructure, qu'il faut résourdre à chaque point de Gauss du maillage. Par ailleurs, pour une résolution optimale, il est nécessaire de calculer l'opérateur tangent associé. A chaque étape de l'algorithme, la résolution de systèmes non-linéaires à l'échelle des phases est en outre nécessaire, ce qui accroît la complexité du calcul. Il est proposé dans cet article une implémentation efficiente de tels schémas d'homogénéisation, où les systèmes non-linéaires sont résolus de manière implicite à toutes les échelles, avec le calcul des opérateurs tangents. Une méthode de condensation statique permet de considérer un grand nombre de phases, et l'implémentation permet également d'utiliser des lois locales arbitraires sur chaque phase. Un exemple de calcul aux éléments finis à l'échelle de la structure est réalisé dans le cadre de la transformation de phases métallurgiques.

Abstract :

This paper presents the development of numerical tools for implementing homogenized constitutive laws based on "mean-field" schemes in nonlinear mechanics. These schemes involve a nonlinear system at the microstructure scale, which must be resolved at each Gauss point of the mesh. Furthermore, for an optimal resolution, the associated tangent operator must be calculated. At each step of the algorithm, the resolution of nonlinear systems at the phase scale is also necessary, which increases the computational complexity. This article proposes an efficient implementation of such homogenization schemes, where resolution algorithms are implicit, at all scales, with the computation of tangent operators. A static condensation method allows to consider a large number of phases, and the implementation also permits to use arbitrary laws on each phase. An example of a finite element computation at the structure scale is carried out within the framework of the transformation of metallurgical phases.

Keywords : non-linear homogenization, mean-field homogenization, implicit algorithm, macroscopic tangent operator, static condensation

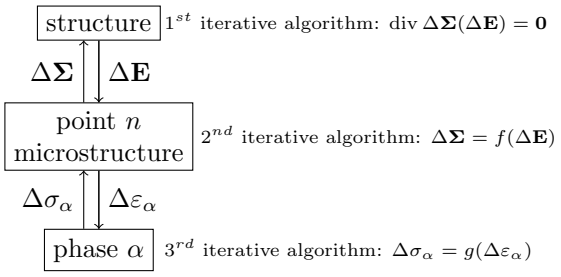


FIGURE 1 – Homogenization procedure from local phase to structure scale, for non-linear behaviours

1 Introduction

Structural analysis is often based on full-field computation which involves a lot of degrees of freedom in a mesh. When the behaviour is non-linear, which is generally the case, it involves a global non-linear problem on the whole structure, whose solution is given by an iterative algorithm of the Newton type.

This algorithm requires the computation of the local stress $\boldsymbol{\sigma}(\boldsymbol{\varepsilon})$ as a function of the local strain increment $\Delta\,\boldsymbol{\varepsilon}$, and it converges optimally when providing the local (consistent) tangent operator $\dfrac{\partial\boldsymbol{\sigma}}{\partial\Delta\,\boldsymbol{\varepsilon}}$ [10].

In addition, most non linear consistutive equations must describe the evolution of a set of internal state variables, which requires to integrate a system of ordinary differential equations *at each Gauss point of the mesh*. For efficiency reasons and, in order to compute the consistent tangent operator, implicit algorithms are preferred.

The *behaviour integration*, is thus not a straightforward step, and its optimization is a key point for the efficiency and the robustness of the global resolution at the structure scale. This is one of the goals of the `MFront` code generator to carry out this integration step [1].

Constitutive laws describing heterogeneous materials, composed by a set of phases, are of particular interest. Their behaviour involves a 'homogenization rule' and the local behaviours of each phase, which also must be integrated. Hence, the global computation involves three different scales (structure, integration point, phases), each of them making intervene an iterative resolution algorithm, as shown on Fig. 1. The general strategy relies on mean-field homogenization, which provides some homogenization rules generally based on analytical expressions. These rules are indeed less time-consuming than full-field computation (FE/FFT, FE²...).

One of the goals of the AnoHonA project, funded by the french ANR Agency, is to address these kinds of behaviours for utilisation in structural codes. However, the implementation of these mean-field schemes may be tedious, (i) depending of the complexity of the scheme, (ii) depending of the complexity of the microstructure and the number of phases, particularly when the behaviour integrations are implicit.

Our contribution here is to provide a numerical framework which permits the user to implement its own homogenization rule (or an existing one) with an arbitrary constitutive law on each phase. We show how to obtain the tangent operator and we use implicit integration. A static condensation method also permits to handle the problem of the number of phases, at the step of the local integration. This paper focuses on well-known schemes (Taylor, Sachs), and we use these schemes in a structural computation.

The paper is organized as follows :

— Section 2 presents the implementation of two non linear homogenization schemes : Taylor and Sachs, with an implicit algorithm, arbitrary local constitutive laws and computation of the tangent operator.
— Section 3 presents the development of `MFront` that allows to use an arbitrary behaviour on each phase.
— Finally, Section 4 is an application at the structure scale, a finite element computation which shows the efficiency of the implementation.

2 Two basic homogenization schemes

2.1 Taylor/Voigt scheme

Taylor/Voigt scheme is a simple homogenization scheme that does not necessitate to solve a non-linear system at the microstructure scale. The only integrations are performed on each phase, and the computation of macroscopic response is straightforward.

Taylor scheme [2] assumes that the strain field on each phase is uniform and equal to the macroscopic strain :

$$\boldsymbol{\varepsilon}(\mathbf{x}) = \mathbf{E} \quad \forall \mathbf{x} \in \Omega_\alpha \tag{1}$$

so that the macroscopic stress is simply given by :

$$\boldsymbol{\Sigma} = \sum_\alpha f_\alpha \boldsymbol{\sigma}_\alpha \tag{2}$$

and the tangent operator is :

$$\frac{\partial \boldsymbol{\Sigma}}{\partial E} = \sum_\alpha f_\alpha \frac{\partial \boldsymbol{\sigma}_\alpha}{\partial \mathbf{E}} = \sum_\alpha f_\alpha \frac{\partial \boldsymbol{\sigma}_\alpha}{\partial \boldsymbol{\varepsilon}_\alpha} \tag{3}$$

These expressions show that computation of both macroscopic stress and consistent tangent operator needs the individual integrations of constitutive laws on each phase. With `MFront` and the formalism given in Section 3, Taylor scheme can hence be implemented with very general local behaviours.

2.2 Sachs scheme

Implementation of Sachs scheme is different from Taylor scheme because it needs to solve a non-linear system at the microstructure scale. Particularly, depending on the number of phases, the jacobian matrix of the non-linear system may be huge. Here, we show how to condensate the system, trough a 'static condensation method' (also known as 'Schür complement') which was proposed for the implementation of Berveiller-Zaoui scheme for a polycrystal containing 240 grains [4]. Our contribution here shows that a very general local law can be used on each phase, contrary to what was presented in previous article.

Sachs scheme [3] is based on the assumption that the stress field on each phase is uniform and equal to the macroscopic stress :

$$\boldsymbol{\sigma}(\mathbf{x}) = \boldsymbol{\Sigma} \quad \forall \mathbf{x} \in \Omega_\alpha. \tag{4}$$

For a 'strain-driven' problem, the macroscopic strain is known and we must have :

$$\mathbf{E} = \sum_{\alpha} f_\alpha \boldsymbol{\varepsilon}_\alpha \tag{5}$$

2.2.1 Non-linear system

We then discretize our time interval, and look for the increments of the variables between t and Δt. Given a variable $\boldsymbol{\varepsilon}$ its increment will be noted $\Delta\boldsymbol{\varepsilon}$.

Hence, the residues of the global non-linear system could then be written as :

$$\begin{aligned} \mathbf{r}_E &= \Delta\mathbf{E} - \sum_{\alpha} f_\alpha \Delta\boldsymbol{\varepsilon}_\alpha \\ \mathbf{r}_{\sigma_\alpha} &= \Delta\boldsymbol{\sigma}_\alpha\left(\boldsymbol{\varepsilon}_\alpha, \Delta\boldsymbol{\varepsilon}_\alpha\right) - \Delta\boldsymbol{\Sigma} \end{aligned} \tag{6}$$

This non-linear system can be solved using for example a Newton-Raphson algorithm.

The consistent tangent operator can also be computed, providing the local tangent operators $\dfrac{\partial\boldsymbol{\sigma}_\alpha}{\partial\boldsymbol{\varepsilon}_\alpha}$ are invertible :

$$\frac{\partial\boldsymbol{\Sigma}}{\partial E} = \left[\sum_{\alpha} f_\alpha \left(\frac{\partial\boldsymbol{\sigma}_\alpha}{\partial\boldsymbol{\varepsilon}_\alpha}\right)^{-1}\right]^{-1}. \tag{7}$$

which is the harmonic mean of the local tangent operators. Of course, the local stress $\Delta\boldsymbol{\sigma}_\alpha\left(\boldsymbol{\varepsilon}_\alpha, \Delta\boldsymbol{\varepsilon}_\alpha\right)$ and the tangent operator $\dfrac{\partial\boldsymbol{\sigma}_\alpha}{\partial\boldsymbol{\varepsilon}_\alpha}$ can be obtained by an *implicit* integration in `MFront`.

2.2.2 Implicit resolution based on static condensation

The Newton-Raphson algorithm follows an iterative procedure in which the unknowns are updated at each iteration, after the resolution of a linear system. Depending on the number of phases, the linear system may contains a high number of unknowns (especially for polycrystalline materials). The idea of the static condensation is to reduce the size of the linear system.

Here, the unknowns are the $\Delta\boldsymbol{\varepsilon}_\alpha$. However, we choose a main unknown, $\Delta\boldsymbol{\varepsilon}_1$, and express the Newton increment of the other unknowns as a function of the increment associated to $\Delta\boldsymbol{\varepsilon}_1$.

We first write the equation giving the increment $\boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_\alpha$ of the unknown $\Delta\boldsymbol{\varepsilon}_\alpha$:

$$-\mathbf{r}_{\sigma_\alpha} = \frac{\partial\boldsymbol{\sigma}_\alpha}{\partial\boldsymbol{\varepsilon}_\alpha} : \boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_\alpha - \frac{\partial\boldsymbol{\sigma}_1}{\partial\boldsymbol{\varepsilon}_1} : \boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_1 \qquad \alpha \geq 2 \tag{8}$$

Assuming that the tangent operator of phase α, $\dfrac{\partial\boldsymbol{\sigma}_\alpha}{\partial\boldsymbol{\varepsilon}_\alpha}$, is invertible, we have :

$$\boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_\alpha = \left(\frac{\partial\boldsymbol{\sigma}_\alpha}{\partial\boldsymbol{\varepsilon}_\alpha}\right)^{-1} : \left(\frac{\partial\boldsymbol{\sigma}_1}{\partial\boldsymbol{\varepsilon}_1} : \boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_1 - \mathbf{r}_{\sigma_\alpha}\right) \tag{9}$$

Then, first equation of system (6) can be used to deduce the increment $\boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_1$. Indeed,

$$-\mathbf{r}_E = -\sum_{\alpha} f_\alpha \boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_\alpha \tag{10}$$

and putting (9) into (10) :

$$-\mathbf{r}_E = -f_1\boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_1 + \sum_{\alpha\geq 2} f_\alpha \mathbf{A}_\alpha - \left(\sum_{\alpha\geq 2} f_\alpha \mathbf{B}_\alpha\right) : \frac{\partial \boldsymbol{\sigma}_1}{\partial \boldsymbol{\varepsilon}_1} : \boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_1 \tag{11}$$

with

$$\mathbf{A}_\alpha = \mathbf{B}_\alpha : \mathbf{r}_{\sigma_\alpha} \qquad \text{and} \quad \mathbf{B}_\alpha = \left(\frac{\partial \boldsymbol{\sigma}_\alpha}{\partial \boldsymbol{\varepsilon}_\alpha}\right)^{-1} \tag{12}$$

The linear system (11) is then solved first, and the increments $\boldsymbol{\delta}\Delta\boldsymbol{\varepsilon}_\alpha$ ($\alpha \geq 2$) are computed afterwards, using the 'decondensation' equation (9).

Here, in the integration of the homogenized behaviour, the local tangent operators on each phase, $\dfrac{\partial \boldsymbol{\sigma}_\alpha}{\partial \boldsymbol{\varepsilon}_\alpha}$, and the increment of the local stress $\Delta\boldsymbol{\sigma}_\alpha\left(\boldsymbol{\varepsilon}_\alpha, \Delta\boldsymbol{\varepsilon}_\alpha\right)$ are needed at each iteration of Newton-Raphson algorithm. These quantities are computed via the utilisation of the keyword `@BehaviourVariable` in `MFront`, which calls the local behaviour laws, and that we will detail in Section 3.

2.3 Computation of the consistent tangent operator

The computation of the tangent operator permits to obtain a quadratic convergence of the iterative algorithm at the structure scale. In the case of Sachs scheme, this computation is straightforward. However, in some other cases, like Cailletaud-Pilvin β rule, we can retrieve it by the mean of the residue which defines an implicit function of the increment of the macroscopic strain $\Delta\mathbf{E}$. Indeed if we note the unknown variables ΔY (a vector of variables), and the residue $\mathbf{R}$ (also a vector of the residues), we can state that

$$\mathbf{R}(\Delta\mathbf{E}, \Delta\mathbf{Y}(\Delta\mathbf{E})) = 0, \qquad \text{and then } \frac{\partial \Delta\mathbf{Y}}{\partial \Delta\mathbf{E}} = -\mathbf{J}^{-1} \cdot \frac{\partial \mathbf{R}}{\partial \Delta\mathbf{E}} \tag{13}$$

where $\mathbf{J} = \dfrac{\partial \mathbf{R}}{\partial \Delta\mathbf{Y}}$ is the jacobian matrix. Then, depending on the unknowns in $\Delta\mathbf{Y}$ and the form of $\dfrac{\partial \mathbf{R}}{\partial \Delta\mathbf{E}}$, the tangent operator may be retrieved (because $\boldsymbol{\Sigma}$ generally depends directly on the components of $\Delta\mathbf{Y}$), as illustrated in the implementation of static condensation for Cailletaud-Pilvin beta rule (see [5]).

3 Implementation within the `MFront` code generator

3.1 The `@BehaviourVariable` keyword

The integration of a constitutive law with `MFront` consists on providing a state, described by internal variables, and an increment of the total strain (called `gradient`) on a time step, to recover a stress (also called `thermodynamic_force`) at the end of the time step. The user has the possibility to give to `MFront` a very general non-linear system which supposes to have a unique solution for computing

the stress at the end of the time step. For an implicit resolution, he must provide the residues of the non-linear system, and the Jacobian matrix of this system (he may also require the numerical computation of this Jacobian). He must also provide the tangent operator. Different types of resolution algorithm can be used : Newton-Raphson, Runge-Kutta, etc. (more details can be found in the documentation of `MFront` [6]). The material behaviour is implemented in a `.mfront` file, and is called a `Behaviour`.

As explained in the above introduction, the implementation of a homogenized constitutive law necessitates to call other constitutive laws, which are related to each local phase of the microstructure. In other words, the implementation of the homogenized `Behaviour` needs to call another `Behaviour`. (In the sequel, we refer to the *calling* behaviour and the *called* `Behaviour`). The interest of this implementation is that the called behaviour can be very general. We hence developed the `MFront` code generator to give this new possibility to the user.

The difficulty of doing this is not only that each called behaviour may have *shared* or *unshared* internal or external variables, but also that the increments of some variables evolve in the calling behaviour (during the resolution algorithm), whereas in the called behaviour, other increments evolve, and the variables of the calling behaviour are used as an initialization for the called behaviour. The declaration of a `Behaviour` inside another can then be written as follows :

```
@BehaviourVariable first_behaviour{
file: "MyBehaviour.mfront",
variables_suffix: "1",
external_names_prefix: "FirstPhase",
store_gradients: false,
store_thermodynamic_forces: true,
shared_material_properties: {".+"},
shared_external_state_variables: {".+"}
};
```

As we can see, the `Behaviour` of the first phase is implemented in an auxiliary file `"MyBehaviour.mfront"`. Some suffixes/prefixes are given for each `StateVariable` of the called behaviour (here, `1` inside the calling behaviour, and `FirstPhase` for an external code). The options `store_gradients` and `store_thermodynamic_forces` allow to store/unstore the `StateVariable`, `AuxiliaryStateVariable`, and stress of the called behaviour. Finally, the `shared_material_properties` and `shared_external_state_variables` are these options which allow to share/unshare some variables with the calling behaviour.

3.2 Implementation of static condensation with Sachs scheme

For simplicity, we consider here two phases, 1 and 2. With the static condensation, the principal variable, ε_1, is declared as `StateVariable`, whereas the other variables ε_2 will be declared `AuxiliaryStateVariable` by the declaration of the local behaviours. The declaration of the called behaviours is done in the calling/homogenized behaviour as explained above. Then, in the core of the `@Integrator` code block, the integration of the called behaviour is written as follows :

```
initialize(b1);
b1.eto=eto1;
b1.deto=deto1;
constexpr auto b1_smflag = TangentOperatorTraits<
MechanicalBehaviourBase::STANDARDSTRAINBASEDBEHAVIOUR
>::STANDARDTANGENTOPERATOR;
b1.integrate(b1_smflag,CONSISTENTTANGENTOPERATOR);
StiffnessTensor Dt1 = b1.getTangentOperator();
```

Here, the initial state is provided by the first two lines. The `gradient` of `b1` is set equal to `deto1`, and the behaviour is integrated by `b1.integrate`. The local stress can be retrieved after by calling `b1.sig`. Last line provides the local tangent operator.

The step of decondensation is implemented in a additional code block, as follows (see eq. (9)) :

```
@ProcessNewCorrection{
b2.deto += iDt2*(b1.sig-b2.sig)+iDt2*Dt1*delta_deto1;
```

where `iDt2` is the inverse of the tangent operator on phase 2. Finally, the consistent tangent operator is implemented in the `TangentOperator` code block (see eq. (7)) :

```
@TangentOperator{
Dt = invert(f*iDt1+(1-f)*iDt2);
};
```

4 Application at the structure scale : a metallurgical phase transformation

In this section we present an application at the structure scale of the homogenization procedure presented, in the context of metallurgical phase transformation.

4.1 Presentation of the problem

We study here the transformation of steel phases in a cooling test. In this test, a circular cylinder is submitted to a fast decrease of temperature from 1000°C to 20°C. There is only an austenite phase at the beginning, and 4 phases are produced during the cooling : bainite, martensite, perlite and ferrite. For simplicity here, we assume that only one phase is produced : martensite. The cylinder has a diameter of $25\,mm$ and a height of $100\,mm$. It is clamped at the top and at the bottom and the other boundaries are free. With the change of temperature, a transformation from austenite to martensite then occurs and the model of Leblond [7] provides us (see subsection 4.1.1) the volume fraction f_α of the different phases as functions of the temperature T. These volume fractions then allow to compute two macroscopic strains : the metallurgical strain $\mathbf{E}^{\mathrm{thm}}$, and the transformation-induced plasticity $\mathbf{E}^{\mathrm{tp}}$. The first one is directly linked to the difference of volume (or compacity) of the martensite phase w. r. t. the austenite phase.

The second is a residual strain which arises when a macroscopic stress $\mathbf{\Sigma}$ (although below the elastic limit of all phases) is applied, and which can be explained by inter-phase interactions (see [8]). The total macroscopic strain is given by

$$\mathbf{E} = \sum_{\alpha} f_\alpha(T)\boldsymbol{\varepsilon}_\alpha + \mathbf{E}^{\mathrm{tp}}(T, \mathbf{\Sigma}) + \mathbf{E}^{\mathrm{thm}}(T) \tag{14}$$

where $\boldsymbol{\varepsilon}_\alpha$ is the average of the local strain on phase α. This macroscopic strain $\mathbf{E}$ is slightly different from the usual average of local strains, but as shown in the following, it has no impact on our computation strategy. The macroscopic stress $\mathbf{\Sigma}$ is as usual,

$$\mathbf{\Sigma} = \sum_{\alpha} f_\alpha \boldsymbol{\sigma}_\alpha \tag{15}$$

where $\boldsymbol{\sigma}_\alpha$ is the average of the local stress on phase α. The task for the mechanician hence consists in computing $\boldsymbol{\varepsilon}_\alpha$ and $\boldsymbol{\sigma}_\alpha$ on each phase. This is done by the homogenization process, that we present in subsection 4.1.3.

4.1.1 Metallurgical model and transformation plasticity

Leblond's model provides the volume fraction of the martensite phase :

$$f_m(T) = 1 - e^{-\beta\langle T_{\mathrm{Ms}} - T\rangle} \tag{16}$$

where for us $T_{\mathrm{Ms}} = 673.15$ K, $\beta = 0.011$ K^{-1} and $\langle\bullet\rangle$ is the positive part of $\bullet$. The metallurgical strain rate is :

$$\dot{\mathbf{E}}^{\mathrm{thm}}(T) = \dot{f}_m\, \varepsilon^{ref}\mathbf{1} \tag{17}$$

where $\varepsilon^{ref} = 0.011$ is the (third of the) volume variation due to the transformation and $\mathbf{1}$ is the second-order identity tensor. The transformation plasticity rate [8] is given by

$$\dot{\mathbf{E}}^{\mathrm{tp}}(T, \mathbf{\Sigma}) = 3\frac{\varepsilon^{ref}}{\sigma_a^y}\dot{f}_m\, \mathbf{S} \tag{18}$$

where σ_a^y is the elastic limit of austenite and $\mathbf{S}$ is the deviatoric part of $\mathbf{\Sigma}$.

4.1.2 Local behaviours

Here, the martensite, the ferrite, the perlite and the bainite have an elasto-plastic behaviour. The austenite has an elasto-viscoplastic behaviour. These behaviours are indeed the one used in [8]. Each local strain is the sum of three terms :

$$\boldsymbol{\varepsilon}_\alpha = \boldsymbol{\varepsilon}_\alpha^{\mathrm{el}} + \boldsymbol{\varepsilon}_\alpha^{\mathrm{th}} + \boldsymbol{\varepsilon}_\alpha^{(\mathrm{v})\mathrm{p}} \tag{19}$$

where Hooke's law provides the elastic strain and $\boldsymbol{\varepsilon}_\alpha^{\mathrm{th}}$ is the thermal strain (for austenite, this strain is null at $T = 1273.15$ K, and for martensite, it is null at $T = 673.15$ K).

The Young modulus is equal to $200-0.17(T-273.15)$ GPa for both phases, and the Poisson ratio equal to 0.3. The thermal expansion coefficient is 15.10^{-6} K^{-1} for martensite and 20.10^{-6} K^{-1} for austenite.

The general form of the elasto-(visco)plastic strain rate is given by the flow rule

$$\dot{\boldsymbol{\varepsilon}}_\alpha^{(\mathrm{v})\mathrm{p}} = \dot{p}_\alpha \, \mathbf{n}_\alpha \qquad \text{with } \mathbf{n}_\alpha = \frac{3}{2}\frac{\boldsymbol{\sigma}_\alpha^d - \mathbf{X}_\alpha}{J(\boldsymbol{\sigma}_\alpha^d - \mathbf{X}_\alpha)} \tag{20}$$

where for all tensor $\mathbf{a}$,

$$J(\mathbf{a}) = \sqrt{\frac{3}{2}\mathbf{a} : \mathbf{a}}, \quad \text{and } \mathbf{a}^d = \mathbf{a} - \frac{1}{3}\mathrm{Tr}(\mathbf{a})\mathbf{1} \tag{21}$$

and $\mathbf{X}_\alpha$ is the thermodynamic force associated to the kinematic hardening. We assume that martensite has an elasto-plastic behaviour with a linear kinematic hardening (Prager) :

$$\begin{aligned}
&\dot{p}_m = \frac{1}{C_m}\left\langle \frac{\boldsymbol{\sigma}_m^d - \mathbf{X}_m}{\sigma_m^y} : \dot{\boldsymbol{\sigma}}_m \right\rangle \qquad \text{if} \qquad J(\boldsymbol{\sigma}_m^d - \mathbf{X}_m) - \sigma_m^y = 0 \\
&\dot{p}_m = 0 \qquad \text{if} \qquad J(\boldsymbol{\sigma}_m^d - \mathbf{X}_m) - \sigma_m^y < 0 \\
&\dot{\mathbf{X}}_m = \frac{2}{3} C_m \, \dot{\boldsymbol{\alpha}}_m \qquad \dot{\boldsymbol{\alpha}}_m = \dot{\boldsymbol{\varepsilon}}_m^{\mathrm{p}} \qquad \mathbf{X}_m(\boldsymbol{\alpha}_m = \mathbf{0}) = 0
\end{aligned} \tag{22}$$

with $C_m = 8$ GPa and $\sigma_m^y = 1 - 0.00012(T - 273.15)$ GPa. And austenite is supposed to have an elasto-viscoplastic behaviours with a kinematic hardening ($\mathbf{X}_a$) and an isotropic hardening (R_a) :

$$\begin{aligned}
&\dot{p}_a = \left\langle \frac{J(\boldsymbol{\sigma}_a^d - \mathbf{X}_a) - R_a - \sigma_a^y}{K_a} \right\rangle^{n_a} \\
&\dot{\mathbf{X}}_a = C_a \, \dot{\boldsymbol{\alpha}}_a \qquad \dot{\boldsymbol{\alpha}}_a = \dot{\boldsymbol{\varepsilon}}_a^{\mathrm{p}} - D_a \, \dot{p}_a \, \boldsymbol{\alpha}_a \qquad \mathbf{X}_a(\boldsymbol{\alpha}_a = \mathbf{0}) = 0 \\
&\dot{R}_a = b_\alpha \, (Q_a - R_a) \, \dot{p}_a \qquad R_a(p_a = 0) = 0
\end{aligned} \tag{23}$$

and we chose the following parameters :

$Q_a = 65$ MPa if $T < 573.15$ K, and $Q_a = 0$ MPa if $T > 573.15$ K ;

$b_a = 0.38$, $C_a = 67(1573.15 - T)$ MPa, $D_a = 960$, $K_a = 820 - 0.0012(1093.15 - T)^2$ MPa.s$^{1/n_a}$, $n_a = 2.86 + 0.007(1273.15 - T)$, and $\sigma_a^y = 200 + 0.4(1273.15 - T)$ MPa.

Remark 1 *All parameters used are inspired from [8], but not strictly identical.*

4.1.3 Homogenization scheme

Different authors attempted to propose a homogenized behaviour for our problem (see [9]). Because the morphology and properties of phases are not well-known, it is a difficult task. Hence, our implementation allows to try different schemes, which is of practical interest for comparing these models to experiments.

For the example here, we first assumed that $\boldsymbol{\varepsilon}_\alpha$ is the same for all phases. Although it is commonly considered as a Taylor model, it must be emphasized that it is sligthly different from what was presented in Section 2.1. Indeed, if the local strain is uniform, it is not equal to the total macroscopic strain $\mathbf{E}$, because of the presence not only of $\mathbf{E}^{\mathrm{thm}}$ but also of $\mathbf{E}^{\mathrm{tp}}$, which depends (in a non-linear way) on the macroscopic stress $\boldsymbol{\Sigma}$. Hence, this scheme *does* necessitate to solve a non-linear problem at the

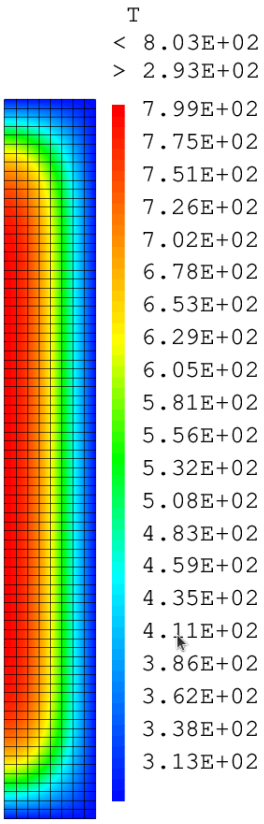


FIGURE 2 – Temperature (in K) in the steel cylinder, at $t = 2.5s$

microstructure scale. There is however only one unknown at this scale : $\boldsymbol{\varepsilon}_\alpha = \mathbf{E} - \mathbf{E}^{\text{thm}} - \mathbf{E}^{\text{tp}}$, and hence there is no need of a static condensation.

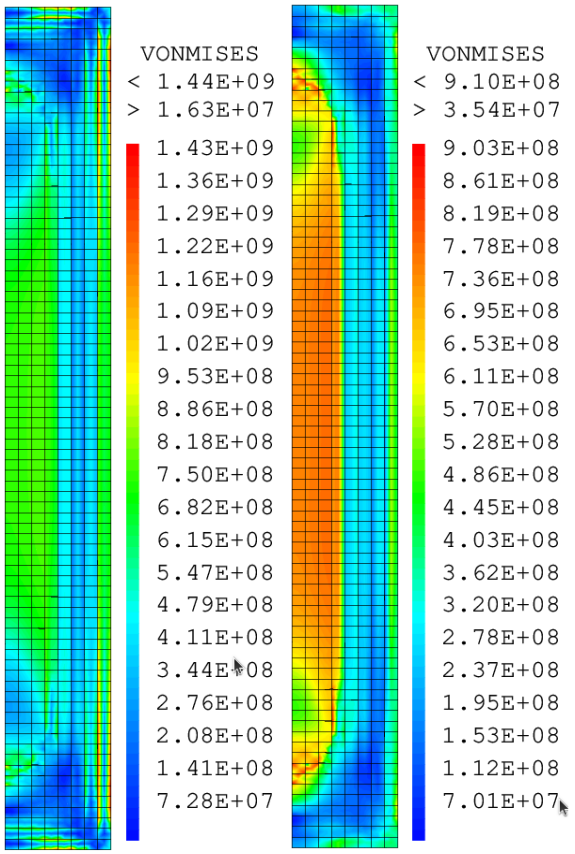
We also tried Sachs scheme, assuming that $\boldsymbol{\sigma}_\alpha$ is uniform and equal to the macroscopic stress $\boldsymbol{\Sigma}$. We conserved eq. (14) for the macroscopic strain. Here again, we obtain a slightly different scheme than previously presented, but the static condensation method can be used.

4.2 Results

The computation was carried out within Cast3M (`www.cast3m.cea.fr`). A thermal computation is carried out first, on an axisymmetric mesh, and the temperature is imposed at each boundary of the cylinder. It decreases from 1000°C to 20°C between $t = 0$ and $t = 1s$, and remains constant at 20°C from $t = 1s$ to $t = 10s$. The temperature is hence non-uniform in the mesh but decreases fastly. Then, a mechanical computation is performed with the precedent temperature field as a given data. This temperature field is shown for $t = 2.5s$ (which is the time also used for the mechanical results) on Fig. 2.

The mesh is composed of 8×80 quadratic quadrangular 2d elements, and the number of time steps is 800 for Sachs scheme (resp. 1600 for Taylor scheme). The procedure `INCREPL` is used at the structure scale. This procedure needs a consistent tangent operator.

The values of the Von Mises stresses are plotted on Fig. 3 for $t = 2.5s$, when the temperature is below 673.15 K in a large part of the steel cylinder. We can hence see the apparition of martensite from the boundaries, that are colder than the center. The colder zone is made of two phases and the behaviour calls each local behaviour, whereas the hotter zone is only made of austenite.



Taylor scheme Sachs scheme

FIGURE 3 – Von Mises stresses in the steel cylinder, at $t = 2.5s$

The transition between the homogeneous and the heterogeneous state is not smooth, however the convergence is obtained at each time step after maximum 5 iterations for Sachs shceme (resp. 9 for Taylor scheme) when the transition occurs.

As expected, in the hotter zone, where only austenite is present, the stresses are the same between the two schemes. However, we can see the difference between the stresses in the colder zone. Taylor scheme predicts a higher stress state (between 300 and 800 MPa) than Sachs scheme (between 200 and 600 MPa), and the difference between the schemes is not negligible. Hence, it would be interesting to try other schemes, for example taking into account the morphology of the phases.

5 Conclusion and outlook

In this contribution, we showed how to implement some mean-field homogenization schemes for structural computation, using implicit algorithms. The implementation allows to use any constitutive law on each phase. The computation of the macroscopic tangent operator is provided for the resolution at the structure scale. A static condensation method permits to handle a large number of phases. All these elements are the bricks that lead to an efficient implementation of complex homogenized constitutive laws. In a further work, schemes that involve second-order moments of the stress/strain fields will be considered.

Acknowledgements

The authors acknowledge the ANR Agency for the financial support of the AnoHonA project (n° AAPG2023), within the framework of which this research was conducted.

Références

[1] T. Helfer, B. Michel, J. M. Proix, M. Salvo, J. Sercombe, M. Casella, Introducing the open-source mfront code generator : Application to mechanical behaviours and material knowledge management within the PLEIADES fuel element modelling platform, Computers & Mathematics with Applications, 70(5) (2015) 994–1023.

[2] G. I. Taylor, Twenty-eighth May lecture to the Institute of Metals, Plastic strain in metals, (1938) 307–324.

[3] G. Sachs, Zur Ableitung einer Fliessbedingung, Z. Ver, Dtsch. Ing., 72 (1928) 734–736.

[4] T. Helfer, C. Gicquel, F. Onimus, R. Masson Implicit integration of the constitutive equations of a polycrystal obtained by the Berveiller-Zaoui homogeneization scheme, 15ème colloque national en calcul des structures, 2022.

[5] https://thelfer.github.io/tfel/web/BetaRule.html

[6] https://thelfer.github.io/tfel/web/documentations.html

[7] J. B. Leblond, J. Devaux, A new kinetic model for anisothermal metallurgical transformations in steels including effect of austenite grain size, Acta metallurgica, 32 (1984) 137–146.

[8] M. Coret, A. Combescure, A mesomodel for the numerical simulation of the multiphasic behavior of materials under anisothermal loading (application to two low-carbon steels), International Journal of Mechanical Sciences, 44 (2002) 1947–1963.

[9] C. Ramard, Étude expérimentale et numérique du soudage multipasse : application à un acier de construction navale, Thèse, Université de Bretagne Sud, 2018.

[10] J. C. Simo, R. L. Taylor, Consistent tangent operators for rate-independent elastoplasticity, Computer Methods in Applied Mechanics and Engineering, 48 (1985) 101–118.